

---

Subject: Geometry input in ROOT (TGeo) format  
Posted by [Volker Friese](#) on Wed, 16 May 2012 12:56:55 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi,

we (CBM) would like eventually to step away from the ASCII geometry format and use ROOT files instead. The main motivations are that the creation of the geometry file seems easier when using TGeo, and that the geometry of a certain subdetector can be viewed and browsed with the ROOT TBrowser standalone, i.e. without invoking the FAIRROOT framework.

Now the documentation of FAIRROOT says that

Quote:Detector geometry in FairRoot can be defined via:

- ASCII files (Hades Geometry interface)
- Root files (TGeoManager Object in ROOT file)

However, I was told by Florian that instead of saving the TGeoManager object to the file, one should save the top volume, which must be of type TGeoVolumeAssembly. If that were true, it will have certain drawbacks:

The top volume of type TGeoVolumeAssembly is not browsable (the geometry is not expanded). Of course, this can be helped by saving both objects, the top volume and the full TGeoManager to the geometry file.

All media defined when creating the geometry file will be ignored by the framework, which will use at runtime those defined in the media.geo file.

Is it possible for the framework to read the TGeoManager instead of the top volume, and add all media defined there to those already defined from media.geo (checking, of course, for double occurrences)?

Best regards,

Volker

---

Subject: Reading from TGeoManager does not work  
Posted by [Volker Friese](#) on Tue, 12 Jun 2012 16:42:18 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

After trying for a while, I concluded that while reading from a TGeoVolume saved in a ROOT file works, reading from a TGeoManager object in the file does not.

There are some straightforward bugs in FairModule::ConstructRootGeometry().

Line 257

```
n=v1->GetNode(0);
```

Since v1 is the volume to be copied (the node below the top node), this sets the node n to its first daughter, i.e. the second node level. Consequently, the wrong transformation matrix is applied when adding v1 to the cave.

Line 259

```
delete NewGeo;
```

This immediately leads to a segmentation fault, since the newly created volume v1 is deleted when deleting the new TGeoManager (MakeCopyVolume adds it to the volume list of the latter).

However, even after correcting that, I did not manage to read in from TGeoManager in a file. I would be interested if anybody did succeed. I personally dislike the must to define all media in one file (media.geo), which quickly gets unreadable. It seems to me, though, that copying objects (volumes, media, materials etc.) from one TGeoManager to another is not foreseen in the ROOT geometry concept - all copy constructors and assignment operators are protected, and there is a lot of usage of the gGeoManager pointer and automatic registration of objects to it.

So I resolved to writing out a TGeoVolume with the detector geometry in it. This works, but one has to take care:

Below the top node (volume), there must be a single volume containing the entire (sub-)detector. This can be a virtual (keep-in) volume or a real one. But only the first daughter of the top volume is considered.

If a medium is assigned whose material name does not correspond to a medium in the TGeoManager (i.e., present in media.geo), the method FairModule::AssignMediumAtImport will create a new medium with the same (material) name but empty properties and assign this to the respective volume. This leads (in my case) to a crash in the transport. It is hard to notice that since there is only a FairLogger output on debug level.

---

Subject: Re: Reading from TGeoManager does not work

Posted by [Mohammad Al-Turany](#) on Thu, 05 Jul 2012 11:46:21 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Hallo Volker,

Quote:There are some straightforward bugs in FairModule::ConstructRootGeometry().

Line 257

```
n=v1->GetNode(0);
```

Since v1 is the volume to be copied (the node below the top node), this sets the node n to its first daughter, i.e. the second node level. Consequently, the wrong transformation matrix is

applied when adding v1 to the cave.

if you look at this same method few lines below you see:

```
TGeoMatrix* M = n->GetMatrix();
```

and when v1 is added to the cave you see:

```
Cave->AddNode(v1,0, M);
```

So this is not really a problem and the right matrix is used.

If a medium is assigned whose material name does not correspond to a medium in the TGeoManager (i.e., present in media.geo), the method FairModule::AssignMediumAtImport will create a new medium with the same (material) name but empty properties and assign this to the respective volume. This leads (in my case) to a crash in the transport. It is hard to notice that since there is only a FairLogger output on debug level.

This is corrected now, if the media is not found in Media file or TGeoManager, it is a FATAL and it exit.

Using the TGeoManager is not working for now and we are trying to solve this problem.

regards,

Mohammad

---

Subject: Re: Reading from TGeoManager does not work  
Posted by [Volker Friese](#) on Thu, 05 Jul 2012 16:30:24 GMT  
[View Forum Message](#) <> [Reply to Message](#)

Quote:if you look at this same method few lines below you see:

```
TGeoMatrix* M = n->GetMatrix();
```

and when v1 is added to the cave you see:

```
Cave->AddNode(v1,0, M);
```

So this is not really a problem and the right matrix is used.

I think that is not correct. Imagine my TGeoManager containing the hierarchy cave - STS - station01. Then

```
NewGeo->cd();
volume=(TGeoVolume*)NewGeo->GetNode(0)->GetDaughter(0)->GetVolume();
will give me the volume of the node STS. This is copied into the volume v1:
v1=volume->MakeCopyVolume(volume->GetShape());
and then the node n is set to
n=v1->GetNode(0);
which is the first daughter node of STS, i.e. station01. So,
TGeoMatrix* M = n->GetMatrix();
is the transformation matrix of station01 into the STS, but what you want to have is the
transformation from STS to cave. This matters only in case of reading from a TGeoManager
(which does not work anyway at the moment); in case of reading from a TGeoVolume, the
implementation is correct.
```

This is corrected now, if the media is not found in Media file or TGeoManager, it is a FATAL and it exit.

Very good, thank you!

Best regards,

Volker

---

---

Subject: Re: Reading from TGeoManager does not work  
Posted by [Mohammad Al-Turany](#) on Fri, 06 Jul 2012 10:07:23 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hallo Volker,

You are right! if we manage to get the TGeoManager import working we have to add a new parameter which specify also which volume from the manger to take and by default the outmost one.

regards,

Mohammad

---